Construction of fractals using formal languages and matrices of attractors

Joëlle Thollot, Eric Tosan

Construction of fractals using formal languages and matrices of attractors

J. THOLLOT and E. TOSAN

LIGIA-LISPI, bât 710
43, boulevard du 11 Novembre 1918
69622 VILLEURBANNE Cedex – France
jthollot@ligia.univ-lyon1.fr

ABSTRACT

LRIFS's (Language-Restricted Iterated Function Systems) generalize the original definition of IFS's (Iterated Function Systems) by providing tools for restricting the sequences of applicable transformations.In this paper, we study an approach of LRIFS's based on matrices and graph theory. This enables us to generate a matrix which elements are attractors.

Key Words: formal language, fractal, iterated function system, language-restricted iterated function system, graph, automaton, matrix, dioïd.

INTRODUCTION

Fractal gemetry was first introduced by MANDELBROT [Mandelbrot83]. However, he didn't give any precise definition of what a fractal is. Thus, several different definitions (not necessarily equivalent) and several conctruction methods have since been tested.

The IFS model [Barnsley88] is particularly interesting due to it's rigourous formalism and it's simplicity : a fractal is encoded by a finite number of contractive transformations. Several authors have tried to generalize this model. Equations systems [Culik II-Dube93][Hart92] and matrices [Peilgen *et al.*92] are a way to define attractor vectors (*i.e.* which elements are non-empty compact sets). Languages accepted by finite-state automaton are a way to define a subset of an IFS attractor [Prusinkiewicz-Hammel92][Berstel-Morcrette89][Berstel-Nait Abdallah89].

We propose an approach based on matrices of the LRIFS model that will enable us to construct a matrix of attractors. We will first summarize the background framework related to LRIFS and finite-state automaton. Then we will show how to associate a matrix to a LRIFS. Finally, we will introduce the definition of our matrix of attractors as well as a construction method.

DEFINITIONS

LRIFS's introduced by PRUSINKIEWICZ and HAMMEL [Prusinkiewicz-Hammel92] generalize
IFS's introduced by BARNSLEY [Barnsley88]. Thus we will give the classical definitions of an IFS and a LRIFS. Then we will introduce a new definition of a LRIFS using a finite-state automaton instead of a language.

Iterated function systems

Let $(\mathcal{X}, d)$ be a complete metric space. Denote by $\mathcal{H}(\mathcal{X})$ the set of all non-empty compacts of $\mathcal{X}$. With the HAUSDORFF distance d_H, $(\mathcal{H}(\mathcal{X}), d_H)$ is a complete metric space.

An IFS is a set $\mathcal{T} = \{T_1, \ldots, T_N\}$ of contractive functions. Then the HUTCHINSON operator defined by :

$$\begin{array}{rccl} F: & \mathcal{P}(\mathcal{X}) & \longrightarrow & \mathcal{P}(\mathcal{X}) \\ & f & \longmapsto & \bigcup_{i=1}^{N} T_i(f) = \mathcal{T} \circ f \end{array}$$

is a contraction on $\mathcal{H}(\mathcal{X})$. Thus this operator has a unique fixed point :

$$f = F(f) = T_1(f) \cup T_2(f) \cup \ldots \cup T_N(f)$$

f is called the attractor of the IFS $\mathcal{T}$ and is denoted by $\mathcal{A}(\mathcal{T})$.

Language-restricted iterated function systems

Barnsley defined an IFS as a set of N functions. Prusinkiewicz defines an IFS as a tuple of functions in order to use formal languages. This enables him to define the alphabet of contraction labels and a language over this alphabet.

A LRIFS is a quadruplet $\mathcal{I}_L = (\mathcal{T}, \Sigma, h, L)$ where :

- $\mathcal{T} = (T_1, \ldots, T_N)$ is a tuple of contractions in $\mathcal{X}$.
- $\Sigma = \{1, \ldots, N\}$ is an alphabet of contraction labels.
- h is a labeling function, defined as :

$$\begin{array}{rccl} h: & \Sigma & \longrightarrow & \mathcal{T} \\ & i & \longmapsto & T_i \end{array}$$

- $L \subset \Sigma^*$ is a language over Σ.

The function h is generalized to languages over Σ using the equations :

$$h(\sigma) = T_{\sigma_1} \circ T_{\sigma_2} \circ \ldots T_{\sigma_k} \text{ if } \sigma = \sigma_1 \sigma_2 \ldots \sigma_k$$

$$h(L) = \{h(\sigma)/\sigma \in L\}$$

$$h(\Sigma^*) = \{h(\sigma)/\sigma \in \Sigma^*\} = \mathcal{T}^*$$

L enables us to construct subsets of transformations and subsets of compacts of the IFS attractor :

$$h(L) \subseteq h(\Sigma^*)$$

$$\mathcal{A}_L(p) = \operatorname{adh}(h(L)(p))$$

$$\mathcal{A}_L(p) \subseteq \mathcal{A}(\mathcal{T}) = \operatorname{adh}(h(\Sigma^*)(p))$$

The set $\mathcal{A}_L(p)$ generally depends on the choice of the starting point p. However, if the language L is postfix extensible (*i.e.* $La \subset L$, the smallest set $\mathcal{A}_L$ does exist and can be found as :

$$\mathcal{A}_L = \operatorname{adh}(h(L)(p_0))$$

where p_0 is the fixed point of the transformation $h(a)$.

Example : We use affine transformations of $\mathbb{R}^2 \times \mathbb{R}^2$. The following notations are used :

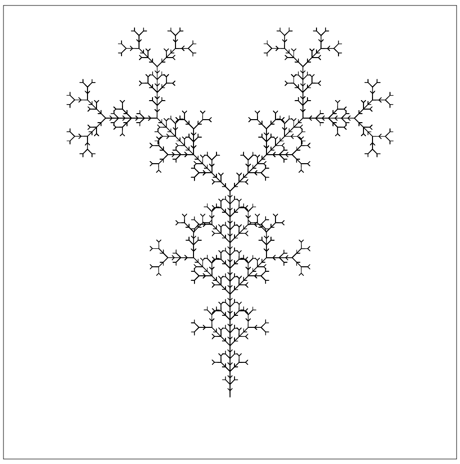
- $T(a, b)$ is a translation by vector (a, b).
- $R(a)$ is a rotation by angle a with respect to the origin of the coordinate system.
- $H(a)$ is a scaling with respect to the origin of the coordinate system.

The following figure shows the attractor of the IFS

$$\mathcal{T} = \{T_1, T_2, T_3, T_4\}$$

where :

$$\begin{array}{rcl} T_1 & = & H(0.5) \\ T_2 & = & T(0, 0.5) \circ H(0.5) \\ T_3 & = & T(0, 1) \circ R(\pi/4) \circ H(0.5) \\ T_4 & = & T(0, 1) \circ R(-\pi/4) \circ H(0.5) \end{array}$$

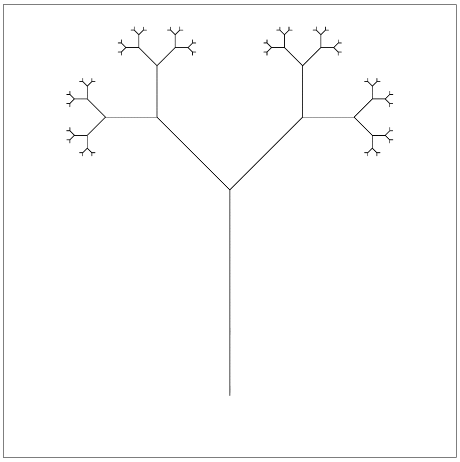


The following figure shows the corresponding attractor of the LRIFS

$$\mathcal{I}_L = (\mathcal{T}, \Sigma, h, L)$$

where :

$$\begin{array}{rcl} \mathcal{T} & = & (T_1, T_2, T_3, T_4) \\ \Sigma & = & \{1, 2, 3, 4\} \\ L & = & \{3, 4\}^* \{1, 2\}^* \end{array}$$



The branching structure of the attractor of the LRIFS is clearly a subset of the original attractor of the IFS.

Finite automaton

In the following, we will only use regular languages as generally admitted in the litterature. We have chosen to work on the graph of the automaton as these languages are accepted by finite automaton. Thus, we will first introduce the definition of a finite automaton and of the graph of an automaton.

A finite automaton is a quintuplet

$$\mathcal{M} = (Q, \Sigma, \delta, Q_I, Q_F)$$

where :

- Q is a finite set of states.
- Σ is an alphabet.
- $\delta : Q \times \Sigma \rightarrow \mathcal{P}(Q)$ is a state transition relation.
- $Q_I \subset Q$ is the set of initial states.
- $Q_F \subset Q$ is the set of final states.

The language accepted by $\mathcal{M}$ is :

$$L(\mathcal{M}) = \{\omega \in \Sigma^* / \exists q \in Q_I : \delta(q, \omega) \subset Q_F\}$$

where $\delta(q, a\omega) = \delta(\delta(q, a), \omega)$ if $a \in \Sigma$ and $\omega \in \Sigma^*$.

We can represent $\mathcal{M}$ as the directed graph $G(\mathcal{M}) = (Q, E)$ with nodes representing states and arcs representing transitions. (The initial and final states will be distinguished by short arrows in figures.)

So we will introduce a novel definition of a LRIFS : a LRIFS is given by a quadruplet $\mathcal{I}_{\mathcal{M}} = (\mathcal{T}, \Sigma, h, \mathcal{M})$.

MATRIX ASSOCIATED WITH A LRIFS

We propose a formalism based on matrices. This is possible because of the properties of the spaces we work on. Thus, we will see what are these properties and how to construct a matrix associated with a LRIFS. More details are given in [Thollot89].

Dioïds

We use the properties of dioïds in order to construct matrices [Gondran-Minoux86].

A dioïd is a triplet $(D, +, \times)$ where :

- D is a set associated with two operations $+$ and $\times$.
- $+$ is commutative and associative.
- $\times$ is distributive over $+$.

Given a dioïd D one can construct and manipulate vectors and matrices which elements are in D. The following triplets are dioïds :

- $(\mathcal{P}(\Sigma^*), \cup, *)$: the set of languages associated with union and concatenation.
- $(\mathcal{P}(h(\Sigma^*)), \cup, \circ)$: the set of transformations sets associated with union and composition.

Moreover, we have the following properties :

$$h(L_1 \cup L_2) = h(L_1) \cup h(L_2)$$

$$h(L_1 * L_2) = h(L_1) \circ h(L_2)$$

Thus, formulae over languages will be the same over sets of transformations.

Matrix associated with a graph of automaton

In $(\mathcal{P}(\Sigma^*), \cup, *)$, we can define the matrix associated with a graph by :

$$A(\mathcal{M}) = (A_{ij})$$

$$A_{ij} = \{a/(q_i, a, q_j) \in E\}$$

Example : Let $\mathcal{M}$ be the automaton that accepts the language $L = \{3,4\}^*\{1,2\}^*$. The graph of $\mathcal{M}$ is shown in Figure 1.

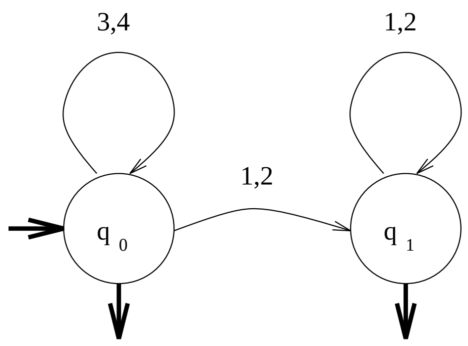


Figure 1

The matrix associated with $\mathcal{M}$ is :

$$A(\mathcal{M}) = \begin{pmatrix} \{3,4\} & \{1,2\} \\ \emptyset & \{1,2\} \end{pmatrix}$$

This matrix does not give the initial and final states. That's why we define two vectors I and F that are respectively the initial and final vectors.

$$I = (I_j) = \begin{cases} \{\epsilon\} & \text{if } q_j \in Q_I \\ \emptyset & \text{if } q_j \notin Q_I \end{cases}$$

$$F = (F_j) = \begin{cases} \{\epsilon\} & \text{if } q_j \in Q_F \\ \emptyset & \text{if } q_j \notin Q_F \end{cases}$$

Proposition : The set of words which lenght is n accepted by $\mathcal{M}$ is :

$$L_n = I^t A(\mathcal{M})^n F$$

Example : Let A be the matrix :

$$A = \begin{pmatrix} \{3,4\}^n & \bigcup_{i+j=n, j\geq 1} \{3,4\}^i\{1,2\}^j \\ \emptyset & \{1,2\}^n \end{pmatrix}$$

The language accepted by the automaton shown in Figure 1 is :

$$L_n = \begin{pmatrix} \{\epsilon\} & \emptyset \end{pmatrix} A \begin{pmatrix} \{\epsilon\} \\ \{\epsilon\} \end{pmatrix}$$

Matrix associated with a LRIFS

Using the matrix associated with an automaton, we can now construct the matrix associated with a LRIFS $\mathcal{I}_\mathcal{M} = (\mathcal{T}, \Sigma, h, \mathcal{M})$ as :

$$H_\mathcal{M} = h(A(\mathcal{M})) = (h(\mathcal{A}_{ij}))$$

The elements of this matrix are the sets of transformations associated with the words of $A(\mathcal{M})$.

Example : The automaton shown in Figure 1 gives the matrix :

$$H_\mathcal{M} = h\begin{pmatrix} \{3,4\} & \{1,2\} \\ \emptyset & \{1,2\} \end{pmatrix}$$

$$H_\mathcal{M} = \begin{pmatrix} \{T_3, T_4\} & \{T_1, T_2\} \\ \emptyset & \{T_1, T_2\} \end{pmatrix}$$

MATRIX OF ATTRACTORS

We have shown in the previous section how to associate a matrix with a graph. We will now see how to associate a matrix of attractor with this matrix. We will introduce an application relating a transformation to a point. Then we will generalize it to sets of transformations and to matrices which elements are sets of transformations.

Case of one transformation

We use a consequence of the fixed point theorem :

Proposition : Let $(\mathcal{X}, d)$ be a complete metric space. Let $\mathcal{S}$ be a set of contractive transformations, stable for $\circ$. Let $T \in \mathcal{S}$ be a contractive transformation. T has a unique fixed point c and we have :

$$\forall p \in \mathcal{X} \quad \lim_{n\to\infty} T^n(p) = c$$

Thus, we can define an application by :

$$\begin{array}{rccc} \alpha : & \mathcal{S} & \longrightarrow & \mathcal{X} \\ & T & \longmapsto & c \end{array}$$

We can also define the following operator :

$$T^\infty = \lim_{n\to\infty} T^n = \mathrm{cst}(c)$$

where $\mathrm{cst}(c)$ is the constant function.

This is a uniform convergence [Gentil92].

Case of a set of transformations

Let $\mathcal{H}(\mathcal{S})$ be the set of all non-empty compacts of $\mathcal{S}$. With the Hausdorff distance,

$(\mathcal{H}(\mathcal{S}), d_H)$ is a complete metric space. Then we have :

$$(\mathcal{T}^{'})^{\infty} = \lim_{n\to\infty} (\mathcal{T}^{'})^n = \mathrm{cst}(\mathcal{A}(\mathcal{T}^{'}))$$

$$\alpha(\mathcal{T}^{'}) = \mathcal{A}(\mathcal{T}^{'})$$

where $\mathcal{T}^{'} = \{T_{\omega_1}, \ldots, T_{\omega_k}\} \in \mathcal{H}(\mathcal{S})$ and $\mathcal{A}(\mathcal{T}^{'})$ is the attractor of the IFS $\mathcal{T}^{'}$.

Example : Consider the LRIFS

$$\mathcal{I}_{\mathcal{M}} = (\mathcal{T}, \Sigma, h, \mathcal{M})$$

where

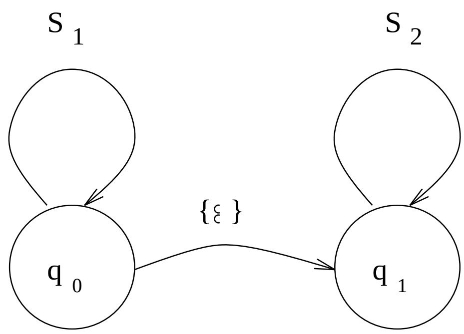
- $\mathcal{T} = \{T_1, \ldots T_8\} = \mathcal{I}_1 \cup \mathcal{I}_2$ where :

$$\mathcal{I}_1 = \{T_1, T_2, T_3, T_4\}$$

$$\mathcal{I}_2 = \{T_5, T_6, T_7, T_8\}$$

$$\begin{aligned}
T_1 &= H(0.5)\\
T_2 &= T(0, 0.5) \circ H(0.5)\\
T_3 &= T(0, 1) \circ R(\pi/4) \circ H(0.5)\\
T_4 &= T(0, 1) \circ R(-\pi/4) \circ H(0.5)\\
T_5 &= H(1/3)\\
T_6 &= T(1, 0) \circ R(\pi/3) \circ H(1/3)\\
T_7 &= T(1, 0) \circ R(\pi/3) \circ T(1, 0)\\
&\quad \circ R(-2\pi/3) \circ H(1/3)\\
T_8 &= T(2, 0) \circ H(1/3)
\end{aligned}$$

- $\Sigma = \{1, \ldots, 8\}$
- $\mathcal{M}$ is given by :



where

$$S_1 = \{1, 2, 3, 4\}$$

$$S_2 = \{5, 6, 7, 8\}$$

Then we have :

$$\alpha(\mathcal{I}_1) = \mathcal{A}_1$$

where $\mathcal{A}_1$ is the attractor of $\mathcal{I}_1$.

$$\alpha(\mathcal{I}_2) = \mathcal{A}_2$$

where $\mathcal{A}_2$ is the attractor of $\mathcal{I}_2$.

Case of a matrix which elements are sets of transformations

Let $M(\mathcal{H}(\mathcal{S}))$ be the set of all matrices which elements are compacts of $\mathcal{S}$. Let d_m be a distance defined by :

$$d_m(A, B) = \max_{1\leq i\leq N, 1\leq j\leq N} d_H(A_{ij}, B_{ij})$$

Then $(M(\mathcal{H}(\mathcal{S})), d_m)$ is a complete metric space, and we have :

$$(H_{\mathcal{M}})^{\infty} = \lim_{n\to\infty} (H_{\mathcal{M}})^n = \mathrm{cst}((\mathcal{A}_{ij}))$$

$$\alpha(H_{\mathcal{M}}) = (\mathcal{A}_{ij})$$

Example :

$$h(A(\mathcal{M})) = H_{\mathcal{M}} = \begin{pmatrix} \mathcal{I}_1 & \{id\} \\ \emptyset & \mathcal{I}_2 \end{pmatrix}$$

$$H_{\mathcal{M}}^k = \begin{pmatrix} \mathcal{I}_1^k & \bigcup_{i+j=k-1} \mathcal{I}_1^i I_2^j \\ \emptyset & \mathcal{I}_2^k \end{pmatrix}$$

$$H_{\mathcal{M}}^{\infty} = \begin{pmatrix} \mathcal{I}_1^{\infty} & \mathcal{I}_1^{\infty} \cup \mathcal{I}_1^{*}\mathcal{I}_2^{\infty} \\ \emptyset & \mathcal{I}_2^{\infty} \end{pmatrix}$$

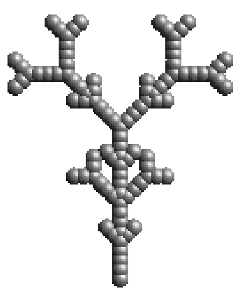
where $\mathcal{I}_1^{*} = \bigcup_k \mathcal{I}_1^k$.

And

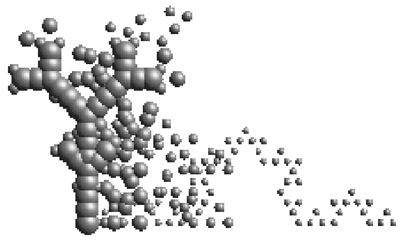
$$\alpha(H_{\mathcal{M}}) = \begin{pmatrix} \mathcal{A}_{11} & \mathcal{A}_{12} \\ \mathcal{A}_{21} & \mathcal{A}_{22} \end{pmatrix}$$

$$\alpha(H_{\mathcal{M}}) = \begin{pmatrix} \mathcal{A}_1 & \mathcal{A}_1 \cup I_1^{*}(\mathcal{A}_2) \\ \emptyset & \mathcal{A}_2 \end{pmatrix}$$

The elements of $\alpha(H_{\mathcal{M}})$ will be :

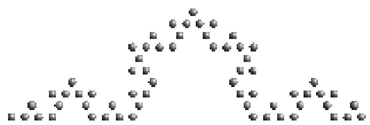


$\mathcal{A}_{11}$



$\mathcal{A}_{12}$

$\mathcal{A}_{21} = \emptyset$



$\mathcal{A}_{22}$

VISUALIZATION

We use the Deterministic Algorithm in order to visualize each element of the matrix [Barnsley88]. We will now give a construction of an attractor and a visualization method using the automaton.

Attractor associated with an automaton

Prusinkiewicz has given a definition of an attractor based on a language. We give a definition of an attractor based on an automaton. This definition is a consequence of the construction of the matrix of attractors.

Definition : The attractor associated with $\mathcal{I}_\mathcal{M} = (\mathcal{T}, \Sigma, h, \mathcal{M})$ is :

$$\mathcal{A}_\mathcal{M} = \bigcup_{q_i \in Q_I, q_j \in Q_F} \mathcal{A}_{ij} \subset \mathcal{A}(\mathcal{T})$$

Relation with languages

The set of all transformations associated with the words of $L(\mathcal{M})$ which lenght is n is :

$$h(L_n) = h(I^t A(\mathcal{M})^n F) = h(I)^t H_\mathcal{M}^n h(F)$$

Then we have :

$$\begin{aligned}
\lim_{n\to\infty} h(L_n) &= h(I)^t(\lim_{n\to\infty} H_\mathcal{M}^n)h(F) \\
&= h(I)^t H_\mathcal{M}^\infty h(F) \\
&= h(I)^t(cst(\mathcal{A}_{ij}))h(F) \\
&= \textstyle\bigcup_{q_i \in Q_I, q_j \in Q_F} cst(\mathcal{A}_{ij}) \\
&= cst(\mathcal{A}_\mathcal{M})
\end{aligned}$$

And so we get :

$$\lim_{n\to\infty} h(L_n) = cst(\mathcal{A}_\mathcal{M})$$

Visualization method

Prusinkiewicz used the escape-time method to visualize the attractor associated with a regular language [Prusinkiewicz-Hammel92]. Culik used the Chaos Game Algorithm to visualize the attractor vector associated with an equations system [Culik II-Dube93]. We use the Deterministic Algorithm to visualize the elements of our matrix of attractors (or an attractor associated with an automaton). This algorithm was also proposed by Culik and used by Mocrette and Berstel [Berstel-Morcrette89][Berstel-Nait Abdallah89].

Let f be a compact (generally the unit ball). Then the sequence $(f_n)_{n\in\mathbb{N}}$ defined by :

$$f_n = h(L_n) \circ f$$

tends to the attractor

$$\mathcal{A}_\mathcal{M} = \bigcup_{q_i \in Q_I, q_j \in Q_F} \mathcal{A}_{ij}$$

With $Q_I = \{i\}$ and $Q_F = \{j\}$ we have :

$$f_n = h(\mathcal{A}_\mathcal{M}^n)_{ij} \circ f$$

This sequence tends to the element $\mathcal{A}_{ij}$ of the matrix of attractors.

Example

Consider the LRIFS

$$\mathcal{I}_{\mathcal{M}} = (\mathcal{T}, \Sigma, h, \mathcal{M})$$

where

- $\mathcal{T} = \{T_1, T_2, T_3, T_4\}$ where :

$$\begin{aligned} T_1 &= H(0.5) \\ T_2 &= T(0.5, 0) \circ H(0.5) \\ T_3 &= T(0, 0.5) \circ H(0.5) \\ T_4 &= T(0.5, 0.5) \circ R(\pi) \circ H(0.5) \end{aligned}$$

- $\Sigma = \{1, \ldots, 4\}$

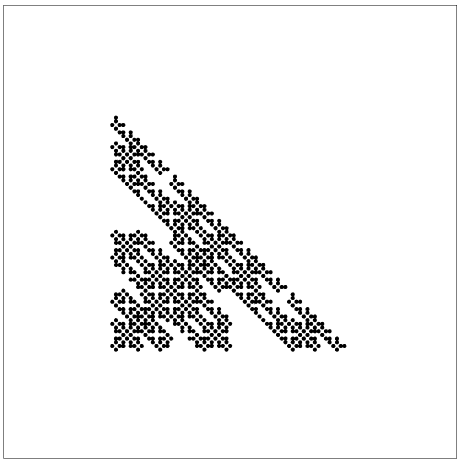
We have chosen the matrix :

$$H_{\mathcal{M}} = \begin{pmatrix} \{T_1, T_4\} & \{T_2, T_3\} \\ \{T_2, T_3\} & \{T_4\} \end{pmatrix}$$

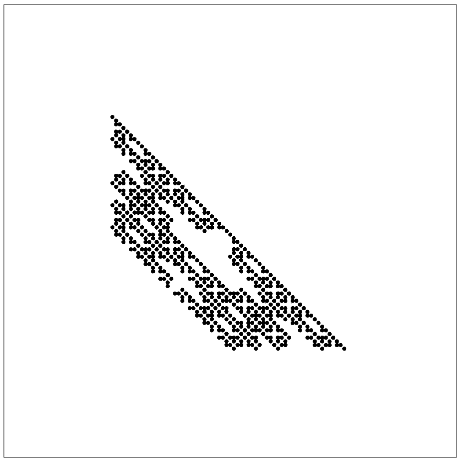
The matrix of attractors corresponding is :

$$\alpha(H_{\mathcal{M}}) = \begin{pmatrix} \mathcal{A}_{11} & \mathcal{A}_{12} \\ \mathcal{A}_{21} & \mathcal{A}_{22} \end{pmatrix}$$

The following figures give the elements of $\alpha(H_{\mathcal{M}})$:



$\mathcal{A}_{11} = \mathcal{A}_{12}$



$\mathcal{A}_{21} = \mathcal{A}_{22}$

CONCLUSION

One can define a matrix of attractors as a limit of a serie of matrices which elements are sets of transformations. Given a matrix of attractors, one can define subsets of attractors associated with a finite automaton. This is a way to combine attractors, by composing elements of the matrix, yielding an interesting approach.

However, further studies are needed in order to :

- Establish relations between the different attractors.
- Develop general visualization algorithms of these attractors.
- Classify the attractors using matrices of languages.
- Establish some "rules" for composing attractors.

References

Barnsley, M.F. 1988. *Fractal Everywhere.* Academic press, INC.

Berstel, J., Morcrette, M. 1989. Compact representation of patterns by finite automata. In *Proceedings of Pixim*, pages 387–395.

Berstel, J., Nait Abdallah, A. 1989. Tétrarbres engendrés par des automates finis. *Langages et algorithmes du graphique, Bigre n° 61-62.*

Culik II, K., Dube, S. 1993. Balancing order and chaos in image generation. *Computer & Graphics*, 17(4), pp. 465–486.

Gentil, C. 1992. *Les Fractales en Synthèse d'Images : le Modèle IFS.* PhD thesis, Université Claude Bernard LYON 1, France.

Gondran, M., Minoux, M. 1986. *Graphes et algorithmes.* Editions Eyrolles.

Hart, J. 1992. The object instancing paradigm for linear fractal modeling. In *Proceedings of Graphics Interface'92.*

Mandelbrot, B.B. 1983. *The Fractal Geometry of Nature.* W.H. Freeman and Co, New York.

Peilgen, H.O., Jurgens, H., Saupe, D. 1992. Encoding images by simple transformations. In *Fractals For The Classroom, New York.*

Prusinkiewicz, P., Hammel, M.S. 1992. Escape-time visualization method for language-restricted iterated function systems. In *Proceedings of Graphics Interface'92.*

Thollot, J. 1989. Langages formels et fractales. Master's thesis, Université Claude Bernard LYON 1, France. Rapport de DEA "Informatique fondamentale".